A screenshot of a computer

Description automatically generated

Imports System.Drawing.Imaging

Imports System.Net.Security

Imports System.Text

Imports Ingres.Client

Public Class Form1

Private Sub btnlogin\_Click(sender As Object, e As EventArgs) Handles btnlogin.Click

If txtuname.Text = "admin" And txtpword.Text = "1234" Then

Me.Hide()

Form2.Show()

ElseIf txtuname.Text = "" And txtpword.Text = "" Then

MsgBox("Please Enter: UserID & Password")

Else

MsgBox("Wrong: 'Username & Password'")

End If

End Sub

Private Sub btnreset\_Click(sender As Object, e As EventArgs) Handles btnreset.Click

txtuname.Text = ""

txtpword.Text = ""

Me.btnreset.Enabled = False

End Sub

Private Sub btnexit\_Click(sender As Object, e As EventArgs) Handles btnexit.Click

Close()

End Sub

End Class

A screenshot of a computer

Description automatically generated

Public Class Form2

Private Sub btnmenu\_Click(sender As Object, e As EventArgs) Handles btnmenu.Click

Form3.Show()

End Sub

Private Sub btnorder\_Click(sender As Object, e As EventArgs) Handles btnorder.Click

Form4.Show()

End Sub

Private Sub btnfinance\_Click(sender As Object, e As EventArgs) Handles btnfinance.Click

Form5.Show()

End Sub

Private Sub btnlogout\_Click(sender As Object, e As EventArgs) Handles btnlogout.Click

Form1.Show()

Me.Close()

End Sub

Private Sub btncalculator\_Click(sender As Object, e As EventArgs) Handles btncalculator.Click

Form6.Show()

End Sub

Private Sub RadMenuItem1\_Click(sender As Object, e As EventArgs)

End Sub

End Class

A screenshot of a computer

Description automatically generated

'imports library "add to reference"

Imports Ingres.Client

Public Class Form3

Dim i As Integer = 0

'Ingress connection

Public IngresConnect As New IngresConnection("host=172.17.243.202;database=ann\_hotelreservation\_db;user id=ingres;password=71xLR!m3&OPH;tz=philippines;")

'Form connection to DB

Private Sub Form3\_Load(sender As Object, e As EventArgs) Handles MyBase.Load

IngConn1 = "host=172.17.243.202;database=ann\_hotelreservation\_db;user id=ingres;password=71xLR!m3&OPH;tz=philippines;"

Call SelectionId()

i = i + 1

txtitemid.Text = CStr(i)

End Sub

#Region "ADD button of menu"

Private Sub btnadd\_Click(sender As Object, e As EventArgs) Handles btnadd.Click

Try

Dim Table As New DataTable

Dim dt As New DataTable

Dim Results As String = "False | Error on saving"

With Table

.Columns.Add("itemid").AutoIncrement = True

.Columns("itemid").AutoIncrementSeed = 1

.Columns.Add("itemname")

.Columns.Add("itemprice")

End With

Table.Rows.Clear()

Table.Rows.Add(txtitemid.Text.ToString.Trim,

txtitemname.Text.ToString.Trim,

txtitemprice.Text.ToString.Trim)

'dtpdob.Value,

Results = CStr(SavingRecords(Table))

Dim Output As Boolean = False

Dim ErrMessage As String = ""

Output = CBool(Split(Results, "|")(0).ToString())

ErrMessage = Split(Results, "|")(1).ToString()

MsgBox(ErrMessage)

Call SelectionId()

Catch ex As Exception

End Try

End Sub

#End Region

'For selection of row in the Datagridview

#Region "DataGridView"

Private Sub DataGridView1\_CellContentClick(sender As Object, e As DataGridViewCellEventArgs) Handles DataGridView1.CellContentClick

Try

Dim table As New DataTable

Dim x As Integer = DataGridView1.CurrentCell.RowIndex

txtitemid.Text = DataGridView1.Rows(x).Cells(0).Value.ToString.Trim

txtitemname.Text = DataGridView1.Rows(x).Cells(1).Value.ToString.Trim

txtitemprice.Text = DataGridView1.Rows(x).Cells(2).Value.ToString.Trim

'dtpdob.Text = DataGridView1.Rows(x).Cells(4).Value

DataGridView1.DataSource = table

Catch ex As Exception

MsgBox(ex.Message)

End Try

End Sub

#End Region

Private Sub btnexit\_Click(sender As Object, e As EventArgs) Handles btnexit.Click

Me.Close()

End Sub

Private Sub btnback\_Click(sender As Object, e As EventArgs) Handles btnback.Click

Form2.Show()

End Sub

Private Sub btnclear\_Click(sender As Object, e As EventArgs) Handles btnclear.Click

txtitemid.Clear()

txtitemname.Clear()

txtitemprice.Clear()

End Sub

'Create Private Sub to call function

'Declare 3 object Table, dta\_record, query to call Db

#Region "MenuDisplay"

Private Sub MenuDisplay()

Try

Dim Table As New DataTable

Dim dta\_record As New DataTable

Dim query As String = "SELECT MAX(itemid) + 1 as itemid FROM Menu"

'"SELECT MAX(itemid) + 1 as itemid FROM Menu"

If OpenFSConn() Then

Dim cmd As New IngresCommand

cmd.Connection = IngresConnect

cmd.CommandText = query

cmd.CommandType = CommandType.Text

cmd.Parameters.Clear()

Dim dta As New IngresDataAdapter(cmd)

dta.Fill(Table)

dta\_record = Table

dta.Dispose()

CloseFSConn()

'txtitemid.Text = cmd.ExecuteScalar().ToString()

End If

DataGridView1.DataSource = Table

DataGridView1.Refresh()

Catch ex As Exception

End Try

End Sub

#End Region

#Region "OpenFSconn & CloseFSconn"

Public Function OpenFSConn() As Boolean

Dim Table As New DataTable

Dim IsOpen As Boolean = False

If IngresConnect.State = ConnectionState.Open Then Return True

Try

IngresConnect.Open()

Catch ex As Exception

End Try

If IngresConnect.State = ConnectionState.Open Then

Dim cm As New IngresCommand("set lockmode session where readlock=nolock", IngresConnect)

cm.ExecuteNonQuery()

IsOpen = True

End If

DataGridView1.DataSource = Table

DataGridView1.Refresh()

Return IsOpen

End Function

Public Sub CloseFSConn()

If IngresConnect.State = ConnectionState.Open Then IngresConnect.Close()

End Sub

#End Region

'Call Function of Menu Display

#Region "ViewDisplay"

Private Sub btnupdate\_Click(sender As Object, e As EventArgs) Handles btnupdate.Click

'Dim table As DataTable

'Dim itemid As Integer = 0

'Dim newitemid As Integer = 0

'table = Selectinfo("sp\_menu\_sel", {})

'table.AcceptChanges()

Call SelectionId()

End Sub

#End Region

'For Delete Function

#Region "Delete"

Private Sub btndelete\_Click(sender As Object, e As EventArgs) Handles btndelete.Click

Dim Table As DataTable

Try

Dim x As Integer = DataGridView1.CurrentCell.RowIndex

Dim itemid As String = DataGridView1.Rows(x).Cells(0).Value.ToString.Trim

Table = IngDataTablemenu("sp\_menu\_del",

{"int|" & itemid})

MsgBox("Successfully Deleted")

Catch ex As Exception

MsgBox(ex.Message)

End Try

End Sub

'For Search function the itemname see the query

#End Region

#Region "Search"

Private Sub btnsearch\_Click(sender As Object, e As EventArgs) Handles btnsearch.Click

Try

Dim Table As New DataTable

Dim dta\_record As New DataTable

Dim query As String = "Select \* FROM Menu WHERE itemname Like '%" + txtsearch.Text + "%'"

If OpenFSConn() Then

Dim cmd As New IngresCommand

cmd.Connection = IngresConnect

cmd.CommandText = query

cmd.CommandType = CommandType.Text

cmd.Parameters.Clear()

Dim dta As New IngresDataAdapter(cmd)

dta.Fill(Table)

dta\_record = Table

dta.Dispose()

CloseFSConn()

End If

DataGridView1.DataSource = Table

DataGridView1.Refresh()

Catch ex As Exception

End Try

End Sub

#End Region

#Region "Selection"

Private Sub SelectionId()

Try

Dim Table As New DataTable

Dim dta\_record As New DataTable

Dim query As String = "Select MAX(itemid) + 1 As itemid FROM Menu"

'DataGridView1.Rows(i).Cells(0).Value.ToString.Trim = txtitemid.Text

If OpenFSConn() Then

Dim cmd As New IngresCommand

cmd.Connection = IngresConnect

cmd.CommandText = query

cmd.CommandType = CommandType.Text

cmd.Parameters.Clear()

Dim dta As New IngresDataAdapter(cmd)

dta.Fill(Table)

dta\_record = Table

dta.Dispose()

CloseFSConn()

End If

DataGridView1.DataSource = Table

DataGridView1.Refresh()

Catch ex As Exception

End Try

End Sub

#End Region

End Class

A screenshot of a computer

Description automatically generated

Imports System.Net.Mime.MediaTypeNames

Imports System.Windows.Forms.VisualStyles.VisualStyleElement

Imports Ingres.Client

Public Class Form4

Public IngresConnect As New IngresConnection("host=172.17.243.202;database=ann\_hotelreservation\_db;user id=ingres;password=71xLR!m3&OPH;tz=philippines;")

Private Sub Form4\_Load(sender As Object, e As EventArgs) Handles MyBase.Load

IngConn1 = "host=172.17.243.202;database=ann\_hotelreservation\_db;user id=ingres;password=71xLR!m3&OPH;tz=philippines;"

Call orderview()

End Sub

'creation of function to call the menu records in Datagridview1

#Region "orderview"

Private Sub orderview()

Try

Dim Table As New DataTable

Dim dta\_record As New DataTable

Dim query As String = "SELECT \* FROM Menu"

If OpenFSConn() Then

Dim cmd As New IngresCommand

cmd.Connection = IngresConnect

cmd.CommandText = query

cmd.CommandType = CommandType.Text

cmd.Parameters.Clear()

Dim dta As New IngresDataAdapter(cmd)

dta.Fill(Table)

dta\_record = Table

dta.Dispose()

CloseFSConn()

End If

DataGridView1.DataSource = Table

DataGridView1.Refresh()

Catch ex As Exception

MsgBox(ex.Message)

End Try

End Sub

#End Region

#Region "Open & Close FSConn"

Public Function OpenFSConn() As Boolean

Dim Table As New DataTable

Dim IsOpen As Boolean = False

If IngresConnect.State = ConnectionState.Open Then Return True

Try

IngresConnect.Open()

Catch ex As Exception

End Try

If IngresConnect.State = ConnectionState.Open Then

Dim cm As New IngresCommand("set lockmode session where readlock=nolock", IngresConnect)

cm.ExecuteNonQuery()

IsOpen = True

End If

DataGridView1.DataSource = Table

DataGridView1.Refresh()

Return IsOpen

End Function

Public Sub CloseFSConn()

If IngresConnect.State = ConnectionState.Open Then IngresConnect.Close()

End Sub

#End Region

#Region "Back & Exit"

Private Sub btnback\_Click(sender As Object, e As EventArgs) Handles btnback.Click

Form2.Show()

Me.Close()

End Sub

Private Sub btnexit\_Click(sender As Object, e As EventArgs) Handles btnexit.Click

Me.Close()

End Sub

#End Region

'computation of quantity

#Region "Quantity compute"

Private Sub txtquantity\_TextChanged(sender As Object, e As EventArgs) Handles txtquantity.TextChanged

If Not IsNumeric(txtquantity.Text) Then

txtquantity.Clear()

End If

txttotal.Text = Val(txtitemprice.Text) \* Val(txtquantity.Text)

txtoriginalprice.Text = Val(txtitemprice.Text) \* Val(txtquantity.Text)

End Sub

#End Region

#Region "Search"

Private Sub btnsearch2\_Click(sender As Object, e As EventArgs) Handles btnsearch2.Click

Try

Dim Table As New DataTable

Dim dta\_record As New DataTable

Dim query As String = "Select \* FROM Menu WHERE itemname Like '%" + txtsearch2.Text + "%'"

If OpenFSConn() Then

Dim cmd As New IngresCommand

cmd.Connection = IngresConnect

cmd.CommandText = query

cmd.CommandType = CommandType.Text

cmd.Parameters.Clear()

Dim dta As New IngresDataAdapter(cmd)

dta.Fill(Table)

dta\_record = Table

dta.Dispose()

CloseFSConn()

End If

DataGridView1.DataSource = Table

DataGridView1.Refresh()

Catch ex As Exception

End Try

End Sub

#End Region

Private Sub btnremove\_Click(sender As Object, e As EventArgs) Handles btnremove.Click

Dim Table As DataTable

Try

Dim x As Integer = DataGridView2.CurrentCell.RowIndex

Dim salesinvoice As String = DataGridView2.Rows(x).Cells(0).Value.ToString.Trim

Table = IngDataTableorderdel("sp\_order\_del",

{"int|" & salesinvoice})

MsgBox("Successfully Deleted")

Catch ex As Exception

MsgBox(ex.Message)

End Try

End Sub

Private Sub DataGridView1\_CellContentClick\_1(sender As Object, e As DataGridViewCellEventArgs) Handles DataGridView1.CellContentClick

Try

Dim table As New DataTable

Dim x As Integer = DataGridView1.CurrentCell.RowIndex

txtSINo.Text = DataGridView1.Rows(x).Cells(0).Value.ToString.Trim

txtitemname.Text = DataGridView1.Rows(x).Cells(1).Value.ToString.Trim

txtitemprice.Text = DataGridView1.Rows(x).Cells(2).Value.ToString.Trim

DataGridView1.DataSource = table

Catch ex As Exception

MsgBox(ex.Message)

End Try

End Sub

Private Sub btndatabase\_Click(sender As Object, e As EventArgs) Handles btnSubmit.Click

Try

Dim Table As New DataTable

Dim Results As String = "False | Error on saving"

With Table

.Columns.Add("salesinvoice")

.Columns.Add("itemname")

.Columns.Add("itemprice")

.Columns.Add("quantity")

.Columns.Add("originalprice")

.Columns.Add("discounted")

.Columns.Add("total")

.Columns.Add("datetime")

End With

Table.Rows.Clear()

Table.Rows.Add(txtSINo.Text.ToString.Trim,

txtitemname.Text.ToString.Trim,

txtitemprice.Text.ToString.Trim,

txtquantity.Text.ToString.Trim,

txtoriginalprice.Text.ToString.Trim,

txtdiscounted.Text.ToString.Trim,

txttotal.Text.ToString.Trim,

dtpdatetime.Value)

Results = CStr(SavingRecords2(Table))

Dim Output As Boolean = False

Dim ErrMessage As String = ""

Output = CBool(Split(Results, "|")(0).ToString())

ErrMessage = Split(Results, "|")(1).ToString()

MsgBox(ErrMessage)

Call order()

Catch ex As Exception

End Try

End Sub

Private Sub order()

Try

Dim Table As New DataTable

Dim dta\_record As New DataTable

Dim query As String = "SELECT \* FROM order"

If OpenFSConn() Then

Dim cmd As New IngresCommand

cmd.Connection = IngresConnect

cmd.CommandText = query

cmd.CommandType = CommandType.Text

cmd.Parameters.Clear()

Dim dta As New IngresDataAdapter(cmd)

dta.Fill(Table)

dta\_record = Table

dta.Dispose()

CloseFSConn()

End If

DataGridView2.DataSource = Table

DataGridView2.Refresh()

Catch ex As Exception

End Try

End Sub

Private Sub btnview\_Click(sender As Object, e As EventArgs) Handles btnview.Click

Call orderview()

End Sub

Private Sub btnclear\_Click(sender As Object, e As EventArgs) Handles btnclear.Click

txtSINo.Clear()

txtitemname.Clear()

txtitemprice.Clear()

txtquantity.Clear()

txtoriginalprice.Clear()

txtdiscounted.Clear()

txttotal.Clear()

End Sub

Private Sub btnregular\_Click(sender As Object, e As EventArgs) Handles btnregular.Click

txttotal.Text = (Val(txtitemprice.Text) \* Val(txtquantity.Text)) \* (1 - 0.2)

txtdiscounted.Text = (Val(txtitemprice.Text) \* Val(txtquantity.Text)) \* (0.2)

End Sub

Private Sub btnsenior\_Click(sender As Object, e As EventArgs) Handles btnsenior.Click

txttotal.Text = (Val(txtitemprice.Text) \* Val(txtquantity.Text)) \* (1 - 0.15)

txtdiscounted.Text = (Val(txtitemprice.Text) \* Val(txtquantity.Text)) \* (0.15)

End Sub

'Private Sub btnemployee\_Click(sender As Object, e As EventArgs)

' txttotal.Text = (Val(txtitemprice.Text) \* Val(txtquantity.Text))

' txtdiscounted.Text = 0

'End Sub

Private Sub DataGridView2\_CellContentClick(sender As Object, e As DataGridViewCellEventArgs) Handles DataGridView2.CellContentClick

Try

Dim x As Integer = DataGridView2.CurrentCell.RowIndex

txtSINo.Text = DataGridView2.Rows(x).Cells(0).Value.ToString().Trim

txtitemname.Text = DataGridView2.Rows(x).Cells(1).Value.ToString().Trim

txtitemprice.Text = DataGridView2.Rows(x).Cells(2).Value.ToString().Trim

txtquantity.Text = DataGridView2.Rows(x).Cells(4).Value.ToString().Trim

txtoriginalprice.Text = DataGridView2.Rows(x).Cells(5).Value.ToString().Trim

txtdiscounted.Text = DataGridView2.Rows(x).Cells(6).Value.ToString().Trim

txttotal.Text = DataGridView2.Rows(x).Cells(7).Value.ToString().Trim

dtpdatetime.Text = DataGridView2.Rows(x).Cells(8).Value

Catch ex As Exception

End Try

End Sub

Private Sub Label1\_Click(sender As Object, e As EventArgs) Handles Label1.Click

End Sub

Private Sub txtoriginalprice\_TextChanged(sender As Object, e As EventArgs) Handles txtoriginalprice.TextChanged

End Sub

End Class

A screenshot of a computer

Description automatically generated

**Imports Ingres.Client**

**Imports System.Data**

**Imports System.Data.SqlClient**

**Imports System.IO**

**Imports System.Net.Security**

**Imports System.Timers**

**Imports System.Windows.Forms.VisualStyles**

**Imports NPOI.XSSF.UserModel**

**Imports Christian\_Auditing\_System.Form5**

**Imports System.ComponentModel**

**Imports NPOI.SS.Formula.Functions**

**Imports Telerik.WinControls.Export**

**Imports System.Reflection**

**Imports iTextSharp.text**

**Imports iTextSharp.text.pdf**

**Public Class Form5**

**Public IngresConnect As New IngresConnection("host=172.17.243.202;database=ann\_hotelreservation\_db;user id=ingres;password=71xLR!m3&OPH;tz=philippines;")**

**Private Sub Form4\_Load(sender As Object, e As EventArgs) Handles MyBase.Load**

**IngConn1 = "host=172.17.243.202;database=ann\_hotelreservation\_db;user id=ingres;password=71xLR!m3&OPH;tz=philippines;"**

**Call Financeorder()**

**End Sub**

**Public Sub New()**

**InitializeComponent()**

**Me.BindDataGridView()**

**End Sub**

**Private Sub BindDataGridView()**

**Dim dt As New DataTable()**

**dt.Columns.AddRange(New DataColumn() {New DataColumn("salesinvoice", GetType(Integer)),**

**New DataColumn("itemname", GetType(String)),**

**New DataColumn("itemprice", GetType(String)),**

**New DataColumn("quantity", GetType(String)),**

**New DataColumn("originalprice", GetType(String)),**

**New DataColumn("discounted", GetType(String)),**

**New DataColumn("total", GetType(String))})**

**Me.DataGridView1.DataSource = dt**

**End Sub**

**Private Sub Button1\_Click(sender As Object, e As EventArgs) Handles btnsearch.Click**

**Try**

**Dim Table As New DataTable**

**Dim dta\_record As New DataTable**

**Dim query As String = "Select \* FROM order WHERE itemname Like '%" + txtsearch.Text + "%'"**

**If OpenFSConn() Then**

**Dim cmd As New IngresCommand**

**cmd.Connection = IngresConnect**

**cmd.CommandText = query**

**cmd.CommandType = CommandType.Text**

**cmd.Parameters.Clear()**

**Dim dta As New IngresDataAdapter(cmd)**

**dta.Fill(Table)**

**dta\_record = Table**

**dta.Dispose()**

**CloseFSConn()**

**End If**

**DataGridView1.DataSource = Table**

**DataGridView1.Refresh()**

**Catch ex As Exception**

**End Try**

**End Sub**

**Private Sub Financeorder()**

**Try**

**Dim Table As New DataTable**

**Dim dta\_record As New DataTable**

**Dim query As String = "SELECT \* FROM Order"**

**If OpenFSConn() Then**

**Dim cmd As New IngresCommand**

**cmd.Connection = IngresConnect**

**cmd.CommandText = query**

**cmd.CommandType = CommandType.Text**

**cmd.Parameters.Clear()**

**Dim dta As New IngresDataAdapter(cmd)**

**dta.Fill(Table)**

**dta\_record = Table**

**dta.Dispose()**

**CloseFSConn()**

**End If**

**DataGridView1.DataSource = Table**

**DataGridView1.Refresh()**

**Catch ex As Exception**

**MsgBox(ex.Message)**

**End Try**

**''configuration for save file dialog**

**'SaveFileDialog1.FileName = ""**

**'SaveFileDialog1.Filter = "PDF (\*.pdf)|\*.pdf"**

**End Sub**

**#Region "Open & CLose FSConn"**

**Public Function OpenFSConn() As Boolean**

**Dim Table As New DataTable**

**Dim IsOpen As Boolean = False**

**If IngresConnect.State = ConnectionState.Open Then Return True**

**Try**

**IngresConnect.Open()**

**Catch ex As Exception**

**End Try**

**If IngresConnect.State = ConnectionState.Open Then**

**Dim cm As New IngresCommand("set lockmode session where readlock=nolock", IngresConnect)**

**cm.ExecuteNonQuery()**

**IsOpen = True**

**End If**

**DataGridView1.DataSource = Table**

**DataGridView1.Refresh()**

**Return IsOpen**

**End Function**

**Public Sub CloseFSConn()**

**If IngresConnect.State = ConnectionState.Open Then IngresConnect.Close()**

**End Sub**

**#End Region**

**Private Sub Button4\_Click(sender As Object, e As EventArgs) Handles btnexport.Click**

**Try**

**Dim worker As New System.ComponentModel.BackgroundWorker**

**AddHandler worker.DoWork, AddressOf ExportLoad**

**AddHandler worker.RunWorkerCompleted, AddressOf ExportCompleted**

**worker.RunWorkerAsync()**

**Catch ex As Exception**

**MsgBox(ex.Message)**

**End Try**

**End Sub**

**Public Structure ResultExporXYZ**

**Dim isSuccess As Boolean**

**Dim hssfwb As XSSFWorkbook**

**End Structure**

**Private Sub ExportLoad(ByVal sender As System.Object, ByVal e As System.ComponentModel.DoWorkEventArgs)**

**Dim Result As New ResultExporXYZ**

**Result.isSuccess = False**

**Try**

**Dim hssfwb = New XSSFWorkbook(NPOI.OpenXml4Net.OPC.OPCPackage.Open(New MemoryStream(My.Resources.template)))**

**Dim formatSheet As XSSFSheet = hssfwb.GetSheet("Sheet1")**

**Dim copyrow As XSSFRow = formatSheet.GetRow(0)**

**Dim begincounter As Integer = 1**

**For X As Integer = 0 To DataGridView1.Rows.Count - 1**

**copyrow.CopyRowTo(begincounter)**

**Dim cellrow As XSSFRow = formatSheet.GetRow(begincounter)**

**cellrow.GetCell(0).SetCellValue(DataGridView1.Rows(X).Cells(0).Value.ToString)**

**cellrow.GetCell(1).SetCellValue(DataGridView1.Rows(X).Cells(1).Value.ToString)**

**cellrow.GetCell(2).SetCellValue(DataGridView1.Rows(X).Cells(2).Value.ToString)**

**cellrow.GetCell(3).SetCellValue(DataGridView1.Rows(X).Cells(3).Value.ToString)**

**cellrow.GetCell(4).SetCellValue(DataGridView1.Rows(X).Cells(4).Value.ToString)**

**cellrow.GetCell(5).SetCellValue(DataGridView1.Rows(X).Cells(5).Value.ToString)**

**cellrow.GetCell(6).SetCellValue(DataGridView1.Rows(X).Cells(6).Value.ToString)**

**cellrow.GetCell(7).SetCellValue(DataGridView1.Rows(X).Cells(7).Value.ToString)**

**begincounter += 1**

**Next**

**Result.hssfwb = hssfwb**

**Result.isSuccess = True**

**Catch ex As Exception**

**MsgBox(ex.Message)**

**End Try**

**e.Result = Result**

**End Sub**

**Private Sub ExportCompleted(ByVal sender As System.Object, ByVal e As System.ComponentModel.RunWorkerCompletedEventArgs)**

**Dim Result As ResultExporXYZ = e.Result**

**If Result.isSuccess = True Then**

**Dim SaveDialog As New SaveFileDialog**

**SaveDialog.Title = "Save Report"**

**SaveDialog.Filter = "Excel Files|\*.xlsx"**

**SaveDialog.FilterIndex = 1**

**If SaveDialog.ShowDialog = DialogResult.OK Then**

**Try**

**Dim xlFileName As String = SaveDialog.FileName**

**Dim xfile As New FileStream(SaveDialog.FileName, FileMode.Create, System.IO.FileAccess.Write)**

**Result.hssfwb.Write(xfile)**

**xfile.Close()**

**Process.Start(SaveDialog.FileName)**

**Catch ex As Exception**

**MsgBox(ex.Message)**

**End Try**

**End If**

**End If**

**End Sub**

**Private Sub Button1\_Click\_1(sender As Object, e As EventArgs) Handles btnview.Click**

**Call Financeorder()**

**End Sub**

**Private Sub Button8\_Click(sender As Object, e As EventArgs) Handles btnback.Click**

**Form2.Show()**

**End Sub**

**Private Sub btnclose\_Click(sender As Object, e As EventArgs) Handles btnclose.Click**

**Me.Close()**

**End Sub**

**Private Sub Button5\_Click(sender As Object, e As EventArgs) Handles btntotaldiscount.Click**

**Dim SUM As Integer**

**For i As Integer = 0 To DataGridView1.RowCount - 1**

**SUM += DataGridView1.Rows(i).Cells(5).Value**

**Next**

**txttotaldiscount.Text = SUM.ToString**

**End Sub**

**Private Sub Button6\_Click(sender As Object, e As EventArgs) Handles btngrandtotal.Click**

**Dim gtot As Integer**

**For i As Integer = 0 To DataGridView1.RowCount - 1**

**gtot += DataGridView1.Rows(i).Cells(6).Value**

**Next**

**txtgrandtotal.Text = gtot.ToString**

**End Sub**

**Private Sub btnsort\_Click(sender As Object, e As EventArgs)**

**DataGridView1.Sort(DataGridView1.Columns(7), ListSortDirection.Ascending)**

**End Sub**

**Private Sub txttotaldiscount\_TextChanged(sender As Object, e As EventArgs) Handles txttotaldiscount.TextChanged**

**End Sub**

**Private Sub btnpdf\_Click(sender As Object, e As EventArgs) Handles btnpdf.Click**

**End Sub**

**'Private Sub btnpdf\_Click(sender As Object, e As EventArgs) Handles btnpdf.Click**

**'Dim Paragraph As New Paragraph**

**'Dim PdfFile As New Document(PageSize.A4, 40, 40, 40, 20)**

**'' PdfFile.AddSubject(TextBox3.Text)**

**'Dim Write As PdfWriter = PdfWriter.GetInstance(PdfFile, New FileStream(TextBox2.Text, FileMode.Create))**

**'PdfFile.Open()**

**'Dim pTotal As New Font(iTextSharp.text.Font.FontFamily.TIMES\_ROMAN, 16, iTextSharp.text.Font.BOLD, BaseColor.BLACK)**

**'Dim pTable As New Font(iTextSharp.text.Font.FontFamily.TIMES\_ROMAN, 12, iTextSharp.text.Font.NORMAL, BaseColor.BLACK)**

**'Paragraph = New Paragraph(New Chunk(TextBox4.Text, pTotal))**

**'Paragraph.Alignment = Element.ALIGN\_CENTER**

**'Paragraph.SpacingAfter = 3.0F**

**'PdfFile.Add(Paragraph)**

**'Dim PdfTable As New PdfPTable(DataGridView1.Columns.Count)**

**'PdfTable.TotalWidth = 500.0F**

**'PdfTable.LockedWidth = True**

**'Dim widths(0 To DataGridView1.Columns.Count - 1) As Single**

**'For i As Integer = 0 To DataGridView1.Columns.Count - 1**

**' widths(i) = 1.0F**

**'Next**

**'PdfTable.SetWidths(widths)**

**'PdfTable.HorizontalAlignment = 0**

**'PdfTable.SpacingBefore = 5.0F**

**'Dim pdfcell As PdfPCell = New PdfPCell**

**'For i As Integer = 0 To DataGridView1.Columns.Count - 1**

**' pdfcell = New PdfPCell(New Phrase(New Chunk(DataGridView1.Columns(i).HeaderText, pTable)))**

**' pdfcell.HorizontalAlignment = PdfPCell.ALIGN\_LEFT**

**' PdfTable.AddCell(pdfcell)**

**'Next**

**'For i As Integer = 0 To DataGridView1.Rows.Count - 2**

**' For j As Integer = 0 To DataGridView1.Columns.Count - 1**

**' pdfcell = New PdfPCell(New Phrase(DataGridView1(j, i).Value.ToString(), pTable))**

**' PdfTable.HorizontalAlignment = PdfPCell.ALIGN\_LEFT**

**' PdfTable.AddCell(pdfcell)**

**' Next**

**'Next**

**'PdfFile.Add(PdfTable)**

**'PdfFile.Close()**

**'MessageBox.Show("PDF format success exported !", "Informations", MessageBoxButtons.OK, MessageBoxIcon.Information)**

**'''Creating iTextSharp Table from the DataTable data**

**''Dim pdfTable As New PdfPTable(DataGridView1.ColumnCount)**

**''pdfTable.DefaultCell.Padding = 3**

**''pdfTable.WidthPercentage = 30**

**''pdfTable.HorizontalAlignment = Element.ALIGN\_LEFT**

**''pdfTable.DefaultCell.BorderWidth = 1**

**'''Adding Header row**

**''For Each column As DataGridViewColumn In DataGridView1.Columns**

**'' Dim cell As New PdfPCell(New Phrase(column.HeaderText))**

**'' 'cell.BackgroundColor = New iTextSharp.text.(240, 240, 240)**

**'' pdfTable.AddCell(cell)**

**''Next**

**'''Adding DataRow**

**''For Each row As DataGridViewRow In DataGridView1.Rows**

**'' For Each cell As DataGridViewCell In row.Cells**

**'' pdfTable.AddCell(cell.Value.ToString())**

**'' Next**

**''Next**

**'''Exporting to PDF**

**''Dim folderPath As String = "C:\Users\cpbobila\Documents\PDF"**

**''If Not Directory.Exists(folderPath) Then**

**'' Directory.CreateDirectory(folderPath)**

**''End If**

**''Using stream As New FileStream(folderPath & "DataGridViewExport.pdf", FileMode.Create)**

**'' Dim pdfDoc As New Document(PageSize.A2, 10.0F, 10.0F, 10.0F, 0.0F)**

**'' PdfWriter.GetInstance(pdfDoc, stream)**

**'' pdfDoc.Open()**

**'' pdfDoc.Add(pdfTable)**

**'' pdfDoc.Close()**

**'' stream.Close()**

**''End Using**

**'End Sub**

**End Class**

A screenshot of a calculator

Description automatically generated

Public Class Form6

Dim Firstnum As Decimal

Dim secondnum As Decimal

Dim Operations As Integer

Dim Operator\_Selector As Boolean = False

Private Sub Button1\_Click(sender As Object, e As EventArgs) Handles Button1.Click

If TextBox1.Text <> "0" Then

TextBox1.Text += "1"

Else

TextBox1.Text = "1"

End If

End Sub

Private Sub Button2\_Click(sender As Object, e As EventArgs) Handles Button2.Click

If TextBox1.Text <> "0" Then

TextBox1.Text += "2"

Else

TextBox1.Text = "2"

End If

End Sub

Private Sub Button3\_Click(sender As Object, e As EventArgs) Handles Button3.Click

If TextBox1.Text <> "0" Then

TextBox1.Text += "3"

Else

TextBox1.Text = "3"

End If

End Sub

Private Sub Button8\_Click(sender As Object, e As EventArgs) Handles Button8.Click

If TextBox1.Text <> "0" Then

TextBox1.Text += "4"

Else

TextBox1.Text = "4"

End If

End Sub

Private Sub Button7\_Click(sender As Object, e As EventArgs) Handles Button7.Click

If TextBox1.Text <> "0" Then

TextBox1.Text += "5"

Else

TextBox1.Text = "5"

End If

End Sub

Private Sub Button6\_Click(sender As Object, e As EventArgs) Handles Button6.Click

If TextBox1.Text <> "0" Then

TextBox1.Text += "6"

Else

TextBox1.Text = "6"

End If

End Sub

Private Sub Button12\_Click(sender As Object, e As EventArgs) Handles Button12.Click

If TextBox1.Text <> "0" Then

TextBox1.Text += "7"

Else

TextBox1.Text = "7"

End If

End Sub

Private Sub Button11\_Click(sender As Object, e As EventArgs) Handles Button11.Click

If TextBox1.Text <> "0" Then

TextBox1.Text += "8"

Else

TextBox1.Text = "8"

End If

End Sub

Private Sub Button10\_Click(sender As Object, e As EventArgs) Handles Button10.Click

If TextBox1.Text <> "0" Then

TextBox1.Text += "9"

Else

TextBox1.Text = "9"

End If

End Sub

Private Sub Button16\_Click(sender As Object, e As EventArgs) Handles Button16.Click

If TextBox1.Text <> "0" Then

TextBox1.Text += "0"

End If

End Sub

Private Sub Button15\_Click(sender As Object, e As EventArgs) Handles Button15.Click

If Not (TextBox1.Text.Contains(".")) Then

TextBox1.Text += "."

End If

End Sub

Private Sub Button14\_Click(sender As Object, e As EventArgs) Handles Button14.Click

TextBox1.Text = "0"

End Sub

Private Sub Button4\_Click(sender As Object, e As EventArgs) Handles Button4.Click

Firstnum = TextBox1.Text

TextBox1.Text = "0"

Operator\_Selector = True

Operations = 1 ' = +

End Sub

Private Sub Button5\_Click(sender As Object, e As EventArgs) Handles Button5.Click

Firstnum = TextBox1.Text

TextBox1.Text = "0"

Operator\_Selector = True

Operations = 2 ' = -

End Sub

Private Sub Button9\_Click(sender As Object, e As EventArgs) Handles Button9.Click

Firstnum = TextBox1.Text

TextBox1.Text = "0"

Operator\_Selector = True

Operations = 3 ' = \*

End Sub

Private Sub Button13\_Click(sender As Object, e As EventArgs) Handles Button13.Click

Firstnum = TextBox1.Text

TextBox1.Text = "0"

Operator\_Selector = True

Operations = 4 ' = /

End Sub

Private Sub Button17\_Click(sender As Object, e As EventArgs) Handles Button17.Click

If Operator\_Selector = True Then

secondnum = TextBox1.Text

If Operations = 1 Then

TextBox1.Text = Firstnum + secondnum

ElseIf Operations = 2 Then

TextBox1.Text = Firstnum - secondnum

ElseIf Operations = 3 Then

TextBox1.Text = Firstnum \* secondnum

Else

If secondnum = 0 Then

TextBox1.Text = "Error!"

Else

TextBox1.Text = Firstnum / secondnum

End If

Operator\_Selector = False

End If

End If

End Sub

End Class

**Using TELERIK UI**

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

**Imports System.IO**

**Imports Ingres.Client**

**Imports NPOI.XSSF.UserModel**

**Imports Telerik.WinControls.UI**

**Imports System.Data**

**Public Class frmCore**

**'Ingres Connection DB**

**Public IngresKonek As New IngresConnection("host=172.17.243.202;database=ann\_hotelreservation\_db;user id=ingres;password=71xLR!m3&OPH;tz=philippines;")**

**'Form Connection to DB**

**Private Sub frmCore\_Load(sender As Object, e As EventArgs) Handles MyBase.Load**

**IngConn1 = "host=172.17.243.202;database=ann\_hotelreservation\_db;user id=ingres;password=71xLR!m3&OPH;tz=philippines;"**

**Call MenuView()**

**Call OrderDisplay()**

**Call FinanceDisplay()**

**Call SalesReport()**

**Try**

**Dim output As New DataTable**

**Dim query As String = "SELECT \* FROM tbl\_menu"**

**Using conn As New IngresConnection(IngConn1)**

**conn.Open()**

**Using comm As New IngresCommand()**

**With comm**

**.Connection = conn**

**.CommandType = CommandType.Text**

**.CommandText = query**

**End With**

**Dim DTAdapter As New IngresDataAdapter(comm)**

**DTAdapter.Fill(output)**

**End Using**

**conn.Dispose()**

**conn.Close()**

**End Using**

**output.Columns(0).ColumnName = "Code"**

**RemoveSortOrder(RadComboBox\_Search\_SalesReport)**

**RadComboBox\_Search\_SalesReport.DataSource = output.Copy**

**RadComboBox\_Search\_SalesReport.BestFitColumns()**

**RadComboBox\_Search\_SalesReport.SelectedIndex = -1**

**RGV\_SalesReport.DataSource = output**

**RGV\_SalesReport.Refresh()**

**Catch ex As Exception**

**MsgBox(ex.Message)**

**End Try**

**End Sub**

**Public Sub RemoveSortOrder(ByVal MCB As RadMultiColumnComboBox)**

**Try**

**For x = 0 To MCB.Columns.Count - 1**

**MCB.EditorControl.MasterTemplate.Columns(x).Sort(RadSortOrder.None, True)**

**Next**

**Catch ex As Exception**

**MsgBox(ex.Message)**

**End Try**

**End Sub**

**'Submit function for menu**

**#Region "Button Submit Menu"**

**Private Sub Iconbtn\_submit\_menu\_Click(sender As Object, e As EventArgs) Handles Iconbtn\_submit\_menu.Click**

**Try**

**Dim Table As New DataTable**

**Dim dt As New DataTable**

**Dim Results As String = "False | Error on saving"**

**With Table**

**.Columns.Add("itemid")**

**.Columns.Add("itemname")**

**.Columns.Add("itemprice")**

**End With**

**Table.Rows.Clear()**

**Table.Rows.Add(RadTextBox\_itemid.Text.ToString.Trim,**

**RadTextBox\_itemname.Text.ToString.Trim,**

**RadTextBox\_itemprice.Text.ToString.Trim)**

**'dtpdob.Value,**

**Results = CStr(SavingRecords(Table))**

**Dim Output As Boolean = False**

**Dim ErrMessage As String = ""**

**Output = CBool(Split(Results, "|")(0).ToString())**

**ErrMessage = Split(Results, "|")(1).ToString()**

**MsgBox(ErrMessage)**

**Call MenuView()**

**Catch ex As Exception**

**End Try**

**End Sub**

**#End Region**

**'To Display records from RadGridView Menu**

**#Region "RadGridView\_Menu"**

**Private Sub Rgv\_Menu\_Click(sender As Object, e As EventArgs) Handles Rgv\_Menu.Click**

**Try**

**Dim table As New DataTable**

**Dim x As Integer = Rgv\_Menu.CurrentCell.RowIndex**

**RadTextBox\_itemid.Text = Rgv\_Menu.Rows(x).Cells(0).Value.ToString.Trim**

**RadTextBox\_itemname.Text = Rgv\_Menu.Rows(x).Cells(1).Value.ToString.Trim**

**RadTextBox\_itemprice.Text = Rgv\_Menu.Rows(x).Cells(2).Value.ToString.Trim**

**'dtpdob.Text = DataGridView1.Rows(x).Cells(4).Value**

**Rgv\_Menu.DataSource = table**

**Catch ex As Exception**

**MsgBox(ex.Message)**

**End Try**

**End Sub**

**#End Region**

**'Create Sub Function for MenuView to call the records from DB - used Query**

**#Region "MenuView"**

**Public Sub MenuView()**

**Try**

**Dim Table As New DataTable**

**Dim dta\_record As New DataTable**

**Dim query As String = "SELECT \* from tbl\_menu"**

**'"SELECT MAX(itemid) + 1 as itemid FROM Menu"**

**If OpenFSConn() Then**

**Dim cmd As New IngresCommand**

**cmd.Connection = IngresKonek**

**cmd.CommandText = query**

**cmd.CommandType = CommandType.Text**

**cmd.Parameters.Clear()**

**Dim dta As New IngresDataAdapter(cmd)**

**dta.Fill(Table)**

**dta\_record = Table**

**dta.Dispose()**

**CloseFSConn()**

**End If**

**Rgv\_Menu.DataSource = Table**

**Rgv\_Menu.Refresh()**

**Catch ex As Exception**

**End Try**

**End Sub**

**#End Region**

**'Connection from MenuView**

**#Region "OpenFSconn & CloseFSconn"**

**Public Function OpenFSConn() As Boolean**

**Dim Table As New DataTable**

**Dim IsOpen As Boolean = False**

**If IngresKonek.State = ConnectionState.Open Then Return True**

**Try**

**IngresKonek.Open()**

**Catch ex As Exception**

**End Try**

**If IngresKonek.State = ConnectionState.Open Then**

**Dim cm As New IngresCommand("set lockmode session where readlock=nolock", IngresKonek)**

**cm.ExecuteNonQuery()**

**IsOpen = True**

**End If**

**Rgv\_Menu.DataSource = Table**

**Rgv\_Menu.Refresh()**

**Return IsOpen**

**End Function**

**Public Sub CloseFSConn()**

**If IngresKonek.State = ConnectionState.Open Then IngresKonek.Close()**

**End Sub**

**#End Region**

**'Search for Menu using itemname**

**#Region "Button for Search of Menu"**

**Private Sub IconBtn\_search\_menu\_Click(sender As Object, e As EventArgs) Handles IconBtn\_search\_menu.Click**

**Try**

**Dim Table As New DataTable**

**Dim dta\_record As New DataTable**

**Dim query As String = "Select \* FROM tbl\_menu WHERE itemname Like '%" + RadTextBox\_search\_menu.Text + "%'"**

**If OpenFSConn() Then**

**Dim cmd As New IngresCommand**

**cmd.Connection = IngresKonek**

**cmd.CommandText = query**

**cmd.CommandType = CommandType.Text**

**cmd.Parameters.Clear()**

**Dim dta As New IngresDataAdapter(cmd)**

**dta.Fill(Table)**

**dta\_record = Table**

**dta.Dispose()**

**CloseFSConn()**

**End If**

**Rgv\_Menu.DataSource = Table**

**Rgv\_Menu.Refresh()**

**Catch ex As Exception**

**End Try**

**End Sub**

**#End Region**

**'Delete Function for Menu**

**#Region "Button for Delete of Menu"**

**Private Sub IconBtn\_delete\_menu\_Click(sender As Object, e As EventArgs) Handles IconBtn\_delete\_menu.Click**

**Dim Table As DataTable**

**Try**

**Dim x As Integer = Rgv\_Menu.CurrentCell.RowIndex**

**Dim itemid As String = Rgv\_Menu.Rows(x).Cells(0).Value.ToString.Trim**

**Table = IngDataTable\_menu("sp\_menu\_del1",**

**{"int|" & itemid})**

**MsgBox("Successfully Deleted")**

**Catch ex As Exception**

**MsgBox(ex.Message)**

**End Try**

**End Sub**

**#End Region**

**Private Sub IconBtn\_refresh\_menu\_Click(sender As Object, e As EventArgs) Handles IconBtn\_refresh\_menu.Click**

**Call MenuView()**

**End Sub**

**'Auto generate of Max number from Menu**

**#Region "Button for Add"**

**Private Sub IconBtn\_add\_menu\_Click(sender As Object, e As EventArgs) Handles IconBtn\_add\_menu.Click**

**Call Generate\_idnum()**

**End Sub**

**#End Region**

**#Region "Generate Number max"**

**Private Sub Generate\_idnum()**

**Dim dta\_idnum As DataTable**

**Dim idnum As Integer = 0**

**Dim newidnum As Integer = 0**

**dta\_idnum = IngDataTableGenNum("sp\_menu\_sel1",**

**{})**

**dta\_idnum.AcceptChanges()**

**For Each row As DataRow In dta\_idnum.Rows**

**If row(0).ToString.Trim = "" Then**

**idnum = 0**

**Else**

**idnum = Int(row(0))**

**End If**

**Next**

**newidnum = idnum + 1**

**RadTextBox\_itemid.Text = newidnum**

**RadTextBox\_itemname.Clear()**

**RadTextBox\_itemprice.Clear()**

**End Sub**

**#End Region**

**Private Sub Radbtn\_exit\_menu\_Click(sender As Object, e As EventArgs) Handles Radbtn\_exit\_menu.Click**

**Me.Close()**

**End Sub**

**'"<------Order Code------->"**

**#Region "Button call Order\_ViewMenu"**

**Private Sub IconBtn\_view\_menu\_Click(sender As Object, e As EventArgs) Handles IconBtn\_view\_menu.Click**

**Call Order\_ViewMenu()**

**End Sub**

**#End Region**

**'Create Sub ViewMenu**

**#Region "Create Sub for View Menu"**

**Public Sub Order\_ViewMenu()**

**Try**

**Dim Table As New DataTable**

**Dim dta\_record As New DataTable**

**Dim query As String = "SELECT \* FROM tbl\_menu"**

**If OpenFSConn() Then**

**Dim cmd As New IngresCommand**

**cmd.Connection = IngresKonek**

**cmd.CommandText = query**

**cmd.CommandType = CommandType.Text**

**cmd.Parameters.Clear()**

**Dim dta As New IngresDataAdapter(cmd)**

**dta.Fill(Table)**

**dta\_record = Table**

**dta.Dispose()**

**CloseFSConn()**

**End If**

**Rgv\_viewmenu\_order.DataSource = Table**

**Rgv\_viewmenu\_order.Refresh()**

**Catch ex As Exception**

**MsgBox(ex.Message)**

**End Try**

**End Sub**

**#End Region**

**'Search Button - Order**

**#Region "Button Search for Order\_ViewMenu"**

**Private Sub IconBtn\_search\_order\_Click(sender As Object, e As EventArgs) Handles IconBtn\_search\_order.Click**

**Try**

**Dim Table As New DataTable**

**Dim dta\_record As New DataTable**

**Dim query As String = "Select \* FROM tbl\_menu WHERE itemname Like '%" + RadTextBox\_search\_order.Text + "%'"**

**If OpenFSConnOrder() Then**

**Dim cmd As New IngresCommand**

**cmd.Connection = IngresKonek**

**cmd.CommandText = query**

**cmd.CommandType = CommandType.Text**

**cmd.Parameters.Clear()**

**Dim dta As New IngresDataAdapter(cmd)**

**dta.Fill(Table)**

**dta\_record = Table**

**dta.Dispose()**

**CloseFSConnOrder()**

**End If**

**Rgv\_Menu.DataSource = Table**

**Rgv\_Menu.Refresh()**

**Catch ex As Exception**

**MsgBox(ex.Message)**

**End Try**

**End Sub**

**#End Region**

**'Connection from Order\_MenuView**

**#Region "OpenFSconnOrder & CloseFSconnOrder"**

**Public Function OpenFSConnOrder() As Boolean**

**Dim Table As New DataTable**

**Dim IsOpen As Boolean = False**

**If IngresKonek.State = ConnectionState.Open Then Return True**

**Try**

**IngresKonek.Open()**

**Catch ex As Exception**

**End Try**

**If IngresKonek.State = ConnectionState.Open Then**

**Dim cm As New IngresCommand("set lockmode session where readlock=nolock", IngresKonek)**

**cm.ExecuteNonQuery()**

**IsOpen = True**

**End If**

**Rgv\_Menu.DataSource = Table**

**Rgv\_Menu.Refresh()**

**Return IsOpen**

**End Function**

**Public Sub CloseFSConnOrder()**

**If IngresKonek.State = ConnectionState.Open Then IngresKonek.Close()**

**End Sub**

**Private Sub IconBtn\_submit\_order\_Click(sender As Object, e As EventArgs) Handles IconBtn\_submit\_order.Click**

**Try**

**Dim Table As New DataTable**

**Dim Results As String = "False | Error on saving"**

**With Table**

**.Columns.Add("SINo")**

**.Columns.Add("itemname")**

**.Columns.Add("itemprice")**

**.Columns.Add("quantity")**

**.Columns.Add("originalprice")**

**.Columns.Add("total")**

**.Columns.Add("discount")**

**.Columns.Add("date")**

**End With**

**Table.Rows.Clear()**

**Table.Rows.Add(RadTextBox\_SINo\_order.Text.ToString.Trim,**

**RadTextBox\_itemname\_order.Text.ToString.Trim,**

**RadTextBox\_itemprice\_order.Text.ToString.Trim,**

**RadTextBox\_quantity\_order.Text.ToString.Trim,**

**RadTextBox\_originalprice\_order.Text.ToString.Trim,**

**RadTextBox\_total\_order.Text.ToString.Trim,**

**RadTextBox\_discount\_order.Text.ToString.Trim,**

**dtpdate.Value)**

**Results = CStr(SavingRecords2(Table))**

**Dim Output As Boolean = False**

**Dim ErrMessage As String = ""**

**Output = CBool(Split(Results, "|")(0).ToString())**

**ErrMessage = Split(Results, "|")(1).ToString()**

**MsgBox(ErrMessage)**

**Catch ex As Exception**

**MsgBox(ex.Message)**

**End Try**

**End Sub**

**#End Region**

**#Region "Button Clear"**

**Private Sub IconBtn\_clear\_order\_Click(sender As Object, e As EventArgs) Handles IconBtn\_clear\_order.Click**

**RadTextBox\_SINo\_order.Clear()**

**RadTextBox\_itemname\_order.Clear()**

**RadTextBox\_itemprice\_order.Clear()**

**RadTextBox\_quantity\_order.Clear()**

**RadTextBox\_originalprice\_order.Clear()**

**RadTextBox\_total\_order.Clear()**

**RadTextBox\_discount\_order.Clear()**

**End Sub**

**#End Region**

**#Region "Computation"**

**Private Sub Iconbtn\_regular\_order\_Click(sender As Object, e As EventArgs) Handles Iconbtn\_regular\_order.Click**

**RadTextBox\_total\_order.Text = (Val(RadTextBox\_itemprice\_order.Text) \* Val(RadTextBox\_quantity\_order.Text)) \* (1 - 0.2)**

**RadTextBox\_discount\_order.Text = (Val(RadTextBox\_itemprice\_order.Text) \* Val(RadTextBox\_quantity\_order.Text)) \* (0.2)**

**End Sub**

**Private Sub IconBtn\_discount\_order\_Click(sender As Object, e As EventArgs) Handles IconBtn\_discount\_order.Click**

**RadTextBox\_total\_order.Text = (Val(RadTextBox\_itemprice\_order.Text) \* Val(RadTextBox\_quantity\_order.Text)) \* (1 - 0.15)**

**RadTextBox\_discount\_order.Text = (Val(RadTextBox\_itemprice\_order.Text) \* Val(RadTextBox\_quantity\_order.Text)) \* (0.15)**

**End Sub**

**Private Sub QuantityCompute(sender As Object, e As EventArgs) Handles RadTextBox\_quantity\_order.TextChanged**

**If Not IsNumeric(RadTextBox\_quantity\_order.Text) Then**

**RadTextBox\_quantity\_order.Clear()**

**End If**

**RadTextBox\_total\_order.Text = Val(RadTextBox\_itemprice\_order.Text) \* Val(RadTextBox\_quantity\_order.Text)**

**RadTextBox\_originalprice\_order.Text = Val(RadTextBox\_itemprice\_order.Text) \* Val(RadTextBox\_quantity\_order.Text)**

**End Sub**

**#End Region**

**#Region "Select into RadGridView-MenuView"**

**Private Sub Rgv\_viewmenu\_order\_Click(sender As Object, e As EventArgs) Handles Rgv\_viewmenu\_order.Click**

**Try**

**Dim table As New DataTable**

**Dim x As Integer = Rgv\_viewmenu\_order.CurrentCell.RowIndex**

**RadTextBox\_SINo\_order.Text = Rgv\_viewmenu\_order.Rows(x).Cells(0).Value.ToString.Trim**

**RadTextBox\_itemname\_order.Text = Rgv\_viewmenu\_order.Rows(x).Cells(1).Value.ToString.Trim**

**RadTextBox\_itemprice\_order.Text = Rgv\_viewmenu\_order.Rows(x).Cells(2).Value.ToString.Trim**

**Rgv\_viewmenu\_order.DataSource = table**

**Catch ex As Exception**

**MsgBox(ex.Message)**

**End Try**

**End Sub**

**#End Region**

**#Region "RadGridView-Order"**

**Private Sub Rgv\_order\_Click(sender As Object, e As EventArgs) Handles Rgv\_order.Click**

**Try**

**Dim x As Integer = Rgv\_order.CurrentCell.RowIndex**

**RadTextBox\_SINo\_order.Text = Rgv\_order.Rows(x).Cells(0).Value.ToString().Trim**

**RadTextBox\_itemname\_order.Text = Rgv\_order.Rows(x).Cells(1).Value.ToString().Trim**

**RadTextBox\_itemprice\_order.Text = Rgv\_order.Rows(x).Cells(2).Value.ToString().Trim**

**RadTextBox\_quantity\_order.Text = Rgv\_order.Rows(x).Cells(3).Value.ToString().Trim**

**RadTextBox\_originalprice\_order.Text = Rgv\_order.Rows(x).Cells(4).Value.ToString().Trim**

**RadTextBox\_total\_order.Text = Rgv\_order.Rows(x).Cells(5).Value.ToString().Trim**

**RadTextBox\_discount\_order.Text = Rgv\_order.Rows(x).Cells(6).Value.ToString().Trim**

**dtpdate.Text = Rgv\_order.Rows(x).Cells(7).Value**

**Catch ex As Exception**

**MsgBox(ex.Message)**

**End Try**

**End Sub**

**#End Region**

**#Region "OrderDisplay"**

**Private Sub OrderDisplay()**

**Try**

**Dim Table As New DataTable**

**Dim dta\_record As New DataTable**

**Dim query As String = "SELECT \* FROM tbl\_order"**

**If OpenFSConn() Then**

**Dim cmd As New IngresCommand**

**cmd.Connection = IngresKonek**

**cmd.CommandText = query**

**cmd.CommandType = CommandType.Text**

**cmd.Parameters.Clear()**

**Dim dta As New IngresDataAdapter(cmd)**

**dta.Fill(Table)**

**dta\_record = Table**

**dta.Dispose()**

**CloseFSConn()**

**End If**

**Rgv\_order.DataSource = Table**

**Rgv\_order.Refresh()**

**Catch ex As Exception**

**End Try**

**End Sub**

**#End Region**

**'"<------Finance Code------->"**

**#Region "Button Search Finance"**

**Private Sub Iconbtn\_search\_finance\_Click(sender As Object, e As EventArgs) Handles Iconbtn\_search\_finance.Click**

**Try**

**Dim Table As New DataTable**

**Dim dta\_record As New DataTable**

**Dim query As String = "Select \* FROM tbl\_order WHERE itemname Like '%" + RadTextBox\_search\_finance.Text + "%'"**

**If OpenFSConn() Then**

**Dim cmd As New IngresCommand**

**cmd.Connection = IngresKonek**

**cmd.CommandText = query**

**cmd.CommandType = CommandType.Text**

**cmd.Parameters.Clear()**

**Dim dta As New IngresDataAdapter(cmd)**

**dta.Fill(Table)**

**dta\_record = Table**

**dta.Dispose()**

**CloseFSConn()**

**End If**

**Rgv\_finance.DataSource = Table**

**Rgv\_finance.Refresh()**

**Catch ex As Exception**

**MsgBox(ex.Message)**

**End Try**

**End Sub**

**#End Region**

**#Region "RadGridView Finance"**

**Private Sub Rgv\_finance\_Click(sender As Object, e As EventArgs) Handles Rgv\_finance.Click**

**Dim x As Integer = Rgv\_finance.CurrentCell.RowIndex**

**RadTextBox\_SINo\_order.Text = Rgv\_finance.Rows(x).Cells(0).Value.ToString().Trim**

**RadTextBox\_itemname\_order.Text = Rgv\_finance.Rows(x).Cells(1).Value.ToString().Trim**

**RadTextBox\_itemprice\_order.Text = Rgv\_finance.Rows(x).Cells(2).Value.ToString().Trim**

**RadTextBox\_quantity\_order.Text = Rgv\_finance.Rows(x).Cells(3).Value.ToString().Trim**

**RadTextBox\_originalprice\_order.Text = Rgv\_finance.Rows(x).Cells(4).Value.ToString().Trim**

**RadTextBox\_total\_order.Text = Rgv\_finance.Rows(x).Cells(5).Value.ToString().Trim**

**RadTextBox\_discount\_order.Text = Rgv\_finance.Rows(x).Cells(6).Value.ToString().Trim**

**dtpdate.Text = Rgv\_finance.Rows(x).Cells(7).Value**

**End Sub**

**#End Region**

**#Region "Finance Display"**

**Private Sub FinanceDisplay()**

**Try**

**Dim Table As New DataTable**

**Dim dta\_record As New DataTable**

**Dim query As String = "SELECT \* FROM tbl\_order"**

**If OpenFSConn() Then**

**Dim cmd As New IngresCommand**

**cmd.Connection = IngresKonek**

**cmd.CommandText = query**

**cmd.CommandType = CommandType.Text**

**cmd.Parameters.Clear()**

**Dim dta As New IngresDataAdapter(cmd)**

**dta.Fill(Table)**

**dta\_record = Table**

**dta.Dispose()**

**CloseFSConn()**

**End If**

**Rgv\_finance.DataSource = Table**

**Rgv\_finance.Refresh()**

**Catch ex As Exception**

**MsgBox(ex.Message)**

**End Try**

**End Sub**

**#End Region**

**#Region "Total Discount"**

**Private Sub IconBtn\_totaldiscount\_finance\_Click(sender As Object, e As EventArgs) Handles IconBtn\_totaldiscount\_finance.Click**

**Dim SUM As Integer**

**For i As Integer = 0 To Rgv\_finance.RowCount - 1**

**SUM += Rgv\_finance.Rows(i).Cells(6).Value**

**Next**

**RadTextBox\_totdiscount\_finance.Text = SUM.ToString**

**End Sub**

**Private Sub Iconbtn\_grandtotal\_finance\_Click(sender As Object, e As EventArgs) Handles Iconbtn\_grandtotal\_finance.Click**

**Dim SUM As Integer**

**For i As Integer = 0 To Rgv\_finance.RowCount - 1**

**SUM += Rgv\_finance.Rows(i).Cells(5).Value**

**Next**

**RadTextBox\_grandtotal\_finance.Text = SUM.ToString**

**End Sub**

**#End Region**

**Private Sub IconBtn\_Refresh\_Click(sender As Object, e As EventArgs) Handles IconBtn\_Refresh.Click**

**Call FinanceDisplay()**

**End Sub**

**Private Sub IconBtn\_export\_finance\_Click(sender As Object, e As EventArgs) Handles IconBtn\_export\_finance.Click**

**Try**

**Dim worker As New System.ComponentModel.BackgroundWorker**

**AddHandler worker.DoWork, AddressOf ExportLoad**

**AddHandler worker.RunWorkerCompleted, AddressOf ExportCompleted**

**worker.RunWorkerAsync()**

**Catch ex As Exception**

**MsgBox(ex.Message)**

**End Try**

**End Sub**

**Private Sub ExportCompleted(ByVal sender As System.Object, ByVal e As System.ComponentModel.RunWorkerCompletedEventArgs)**

**Dim Result As ResultExporXYZ = e.Result**

**If Result.isSuccess = True Then**

**Dim SaveDialog As New SaveFileDialog**

**SaveDialog.Title = "Save Report"**

**SaveDialog.Filter = "Excel Files|\*.xlsx"**

**SaveDialog.FilterIndex = 1**

**If SaveDialog.ShowDialog = DialogResult.OK Then**

**Try**

**Dim xlFileName As String = SaveDialog.FileName**

**Dim xfile As New FileStream(SaveDialog.FileName, FileMode.Create, System.IO.FileAccess.Write)**

**Result.hssfwb.Write(xfile)**

**xfile.Close()**

**Process.Start(SaveDialog.FileName)**

**Catch ex As Exception**

**MsgBox(ex.Message)**

**End Try**

**End If**

**End If**

**End Sub**

**Public Structure ResultExporXYZ**

**Dim isSuccess As Boolean**

**Dim hssfwb As XSSFWorkbook**

**End Structure**

**Private Sub ExportLoad(ByVal sender As System.Object, ByVal e As System.ComponentModel.DoWorkEventArgs)**

**Dim Result As New ResultExporXYZ**

**Result.isSuccess = False**

**Try**

**Dim hssfwb = New XSSFWorkbook(NPOI.OpenXml4Net.OPC.OPCPackage.Open(New MemoryStream(My.Resources.template1)))**

**Dim formatSheet As XSSFSheet = hssfwb.GetSheet("Sheet1")**

**Dim copyrow As XSSFRow = formatSheet.GetRow(0)**

**Dim begincounter As Integer = 1**

**For X As Integer = 0 To Rgv\_finance.Rows.Count - 1**

**copyrow.CopyRowTo(begincounter)**

**Dim cellrow As XSSFRow = formatSheet.GetRow(begincounter)**

**cellrow.GetCell(0).SetCellValue(Rgv\_finance.Rows(X).Cells(0).Value.ToString)**

**cellrow.GetCell(1).SetCellValue(Rgv\_finance.Rows(X).Cells(1).Value.ToString)**

**cellrow.GetCell(2).SetCellValue(Rgv\_finance.Rows(X).Cells(2).Value.ToString)**

**cellrow.GetCell(3).SetCellValue(Rgv\_finance.Rows(X).Cells(3).Value.ToString)**

**cellrow.GetCell(4).SetCellValue(Rgv\_finance.Rows(X).Cells(4).Value.ToString)**

**cellrow.GetCell(5).SetCellValue(Rgv\_finance.Rows(X).Cells(5).Value.ToString)**

**cellrow.GetCell(6).SetCellValue(Rgv\_finance.Rows(X).Cells(6).Value.ToString)**

**cellrow.GetCell(7).SetCellValue(Rgv\_finance.Rows(X).Cells(7).Value.ToString)**

**begincounter += 1**

**Next**

**Result.hssfwb = hssfwb**

**Result.isSuccess = True**

**Catch ex As Exception**

**MsgBox(ex.Message)**

**End Try**

**e.Result = Result**

**End Sub**

**Private Sub SalesReport()**

**Try**

**Dim Table As New DataTable**

**Dim dta\_record As New DataTable**

**Dim query As String = "SELECT a.itemid, a.itemname, a.itemprice, b.quantity, b.originalprice, b.total, b.discount, b.date**

**FROM tbl\_menu a LEFT JOIN tbl\_order b ON a.itemid = b.sino"**

**If OpenFSConn() Then**

**Dim cmd As New IngresCommand**

**cmd.Connection = IngresKonek**

**cmd.CommandText = query**

**cmd.CommandType = CommandType.Text**

**cmd.Parameters.Clear()**

**Dim dta As New IngresDataAdapter(cmd)**

**dta.Fill(Table)**

**dta\_record = Table**

**dta.Dispose()**

**CloseFSConn()**

**End If**

**RGV\_SalesReport.DataSource = Table**

**RGV\_SalesReport.Refresh()**

**Catch ex As Exception**

**MsgBox(ex.Message)**

**End Try**

**End Sub**

**Private Sub RGV\_SalesReport\_Click(sender As Object, e As EventArgs) Handles RGV\_SalesReport.Click**

**Dim x As Integer = RGV\_SalesReport.CurrentCell.RowIndex**

**RadTextBox\_itemid.Text = RGV\_SalesReport.Rows(x).Cells(0).Value.ToString().Trim**

**RadTextBox\_itemname.Text = RGV\_SalesReport.Rows(x).Cells(1).Value.ToString().Trim**

**RadTextBox\_itemprice.Text = RGV\_SalesReport.Rows(x).Cells(2).Value.ToString().Trim**

**RadTextBox\_quantity\_order = RGV\_SalesReport.Rows(x).Cells(3).Value**

**RadTextBox\_originalprice\_order = RGV\_SalesReport.Rows(x).Cells(4).Value**

**RadTextBox\_total\_order = RGV\_SalesReport.Rows(x).Cells(5).Value**

**RadTextBox\_discount\_order = RGV\_SalesReport.Rows(x).Cells(6).Value**

**End Sub**

**Private Sub IconButton5\_Click(sender As Object, e As EventArgs) Handles IconBtn\_Search\_SalesReport.Click**

**Dim Table As New DataTable**

**Dim query As String = "SELECT a.itemid, a.itemname, a.itemprice, b.quantity, b.originalprice, b.total, b.discount, b.date**

**FROM tbl\_menu a LEFT JOIN tbl\_order b ON a.itemid = b.sino**

**WHERE a.itemid Like '%" + RadComboBox\_Search\_SalesReport.Text + "%'"**

**Try**

**If OpenFSConn() Then**

**Dim cmd As New IngresCommand**

**cmd.Connection = IngresKonek**

**cmd.CommandText = query**

**cmd.CommandType = CommandType.Text**

**cmd.Parameters.Clear()**

**Dim dta As New IngresDataAdapter(cmd)**

**dta.Fill(Table)**

**dta\_record = Table**

**dta.Dispose()**

**CloseFSConn()**

**End If**

**RGV\_SalesReport.DataSource = Table**

**RGV\_SalesReport.Refresh()**

**Catch ex As Exception**

**MsgBox(ex.Message)**

**End Try**

**End Sub**

**Private Sub IconBtn\_LoadChart\_SalesReport\_Click(sender As Object, e As EventArgs, dr As DataTable) Handles IconBtn\_LoadChart\_SalesReport.Click**

**Try**

**Dim output As New DataTable**

**Dim Table As New DataTable**

**Dim query As String = "SELECT total, date FROM tbl\_order "**

**If OpenFSConn() Then**

**Dim cmd As New IngresCommand**

**cmd.Connection = IngresKonek**

**cmd.CommandText = query**

**cmd.CommandType = CommandType.Text**

**cmd.Parameters.Clear()**

**Dim dta As New IngresDataAdapter(cmd)**

**dta.Fill(Table)**

**dta\_record = Table**

**dta.Dispose()**

**CloseFSConn()**

**End If**

**'RadChart\_SalesReport "date".XValueMember = "date"**

**'RadChart\_SalesReport "total".YValueMember = "total"**

**RGV\_SalesReport.Rows.Clear()**

**RadChart\_SalesReport.DataSource = output**

**RadChart\_SalesReport.Refresh()**

**Catch ex As Exception**

**MsgBox(ex.Message)**

**End Try**

**End Sub**

**Private Sub RadButton1\_Click(sender As Object, e As EventArgs)**

**Me.Close()**

**End Sub**

**End Class**

**MODULE**

**Imports Ingres.Client**

**Module Module1**

**Public IngConn1 As String**

**Dim ConnIngres As New IngresConnection**

**Dim ComIngres As New IngresCommand**

**Dim dr As New IngresDataAdapter**

**Dim OutputTable As New DataTable**

**'For Submit connection of SavingRecords "Submit records from the DB(Menu)"**

**#Region "Saving Records (Menu)"**

**Public Function SavingRecords(ByVal table As DataTable) As String**

**Dim Output As Boolean = False**

**Dim ErrMessage As String = "Error on saving person."**

**Dim OracleIntegIngresConn As New IngresConnection**

**OracleIntegIngresConn.ConnectionString = IngConn1**

**Dim OracleIntegIngresComm As New IngresCommand**

**Dim OracleIntegIngresTrans\_JournalEntry As IngresTransaction = Nothing**

**Dim OracleIntegIngresDReader As IngresDataReader**

**Dim IngFields() As String = {}**

**Try**

**OracleIntegIngresConn.Open()**

**ExecuteReadLockSession(OracleIntegIngresConn)**

**OracleIntegIngresTrans\_JournalEntry = OracleIntegIngresConn.BeginTransaction**

**OracleIntegIngresComm.Connection = OracleIntegIngresConn**

**OracleIntegIngresComm.Transaction = OracleIntegIngresTrans\_JournalEntry**

**For Each items In table.AsEnumerable()**

**With OracleIntegIngresComm**

**.Parameters.Clear()**

**.CommandText = "sp\_menu\_ins"**

**.CommandType = Data.CommandType.StoredProcedure**

**End With**

**IngFields = {"int|" & items.ItemArray(0).ToString.Trim,**

**"var|" & items.ItemArray(1).ToString.Trim,**

**"var|" & items.ItemArray(2).ToString.Trim}**

**'"dte|" & items.ItemArray(4),**

**SetParameters(OracleIntegIngresComm, IngFields)**

**OracleIntegIngresDReader = OracleIntegIngresComm.ExecuteReader**

**If Not OracleIntegIngresDReader.IsClosed Then OracleIntegIngresDReader.Close()**

**Next**

**OracleIntegIngresTrans\_JournalEntry.Commit()**

**Output = True**

**ErrMessage = "New Information has been saved."**

**Catch ex As Exception**

**If ErrMessage = "Error on saving Information." Then**

**ErrMessage = "Error on saving Information." & ex.Message**

**End If**

**OracleIntegIngresTrans\_JournalEntry.Rollback()**

**Finally**

**OracleIntegIngresConn.Close()**

**OracleIntegIngresConn.Dispose()**

**OracleIntegIngresComm.Dispose()**

**End Try**

**Return Output & "|" & ErrMessage**

**End Function**

**Public Sub ExecuteReadLockSession(ByVal IngConn1 As IngresConnection)**

**Dim IngresCm As New IngresCommand("set lockmode session where readlock=nolock", IngConn1)**

**If IngConn1.State = ConnectionState.Open Then**

**IngresCm.ExecuteNonQuery()**

**End If**

**IngresCm.Dispose()**

**End Sub**

**Public Function SetParameters(ByRef Input As IngresCommand, ByVal ParamArray IngFields() As String) As Boolean**

**Dim Result As Boolean = True**

**Try**

**For i = 0 To UBound(IngFields)**

**Dim StrDataType As String = IngFields(i).Split(CChar("|")).GetValue(0).ToString**

**Dim StrValue As String = IngFields(i).Split(CChar("|")).GetValue(1).ToString**

**Select Case StrDataType.ToUpper**

**Case Is = "INT"**

**Input.Parameters.Add("Parameter" & i, IngresType.Int).Value = CInt(StrValue)**

**Case Is = "VAR"**

**Input.Parameters.Add("Parameter" & i, IngresType.VarChar).Value = StrValue**

**Case Is = "CHR"**

**Input.Parameters.Add("Parameter" & i, IngresType.Char).Value = StrValue**

**Case Is = "DTE"**

**Input.Parameters.Add("Parameter" & i, IngresType.Date).Value = CDate(StrValue)**

**Case Is = "MON"**

**Input.Parameters.Add("Parameter" & i, IngresType.Double).Value = CDbl(StrValue)**

**End Select**

**Next**

**Catch ex As Exception**

**Result = False**

**End Try**

**Return Result**

**End Function**

**#End Region**

**'For Submit connection of SavingRecords2 "Submit records from the DB(Order)"**

**#Region "Saving Records (Order)"**

**Public Function SavingRecords2(ByVal table As DataTable) As String**

**Dim Output As Boolean = False**

**Dim ErrMessage As String = "Error on saving person."**

**Dim OracleIntegIngresConn As New IngresConnection**

**OracleIntegIngresConn.ConnectionString = IngConn1**

**Dim OracleIntegIngresComm As New IngresCommand**

**Dim OracleIntegIngresTrans\_JournalEntry As IngresTransaction = Nothing**

**Dim OracleIntegIngresDReader As IngresDataReader**

**Dim IngFields() As String = {}**

**Try**

**OracleIntegIngresConn.Open()**

**ExecuteReadLockSession(OracleIntegIngresConn)**

**OracleIntegIngresTrans\_JournalEntry = OracleIntegIngresConn.BeginTransaction**

**OracleIntegIngresComm.Connection = OracleIntegIngresConn**

**OracleIntegIngresComm.Transaction = OracleIntegIngresTrans\_JournalEntry**

**For Each items In table.AsEnumerable()**

**With OracleIntegIngresComm**

**.Parameters.Clear()**

**.CommandText = "sp\_order\_ins1"**

**.CommandType = Data.CommandType.StoredProcedure**

**End With**

**IngFields = {"int|" & items.ItemArray(0).ToString.Trim,**

**"var|" & items.ItemArray(1).ToString.Trim,**

**"var|" & items.ItemArray(2).ToString.Trim,**

**"int|" & items.ItemArray(3).ToString.Trim,**

**"var|" & items.ItemArray(4).ToString.Trim,**

**"var|" & items.ItemArray(5).ToString.Trim,**

**"var|" & items.ItemArray(6).ToString.Trim,**

**"dte|" & items.ItemArray(7)}**

**SetParameters(OracleIntegIngresComm, IngFields)**

**OracleIntegIngresDReader = OracleIntegIngresComm.ExecuteReader**

**If Not OracleIntegIngresDReader.IsClosed Then OracleIntegIngresDReader.Close()**

**Next**

**OracleIntegIngresTrans\_JournalEntry.Commit()**

**Output = True**

**ErrMessage = "New Information has been saved."**

**Catch ex As Exception**

**If ErrMessage = "Error on saving Information." Then**

**ErrMessage = "Error on saving Information." & ex.Message**

**End If**

**OracleIntegIngresTrans\_JournalEntry.Rollback()**

**Finally**

**OracleIntegIngresConn.Close()**

**OracleIntegIngresConn.Dispose()**

**OracleIntegIngresComm.Dispose()**

**End Try**

**Return Output & "|" & ErrMessage**

**End Function**

**#End Region**

**#Region "Delete function for Menu"**

**Public Function IngDataTable\_menu(ByVal ProcedureName As String, ByVal ParamArray IngFields() As String) As DataTable**

**Dim ConnIngres As New IngresConnection**

**Dim ComIngres As New IngresCommand**

**Dim dr As New IngresDataAdapter**

**Dim OutputTable As New DataTable**

**Try**

**ConnIngres.ConnectionString = IngConn1**

**ConnIngres.Open()**

**ExecuteReadLockSession(ConnIngres)**

**With ComIngres**

**.Connection = ConnIngres**

**.Parameters.Clear()**

**.CommandText = "sp\_menu\_del1"**

**.CommandType = Data.CommandType.StoredProcedure**

**End With**

**For i = 0 To UBound(IngFields)**

**Dim StrDataType As String = IngFields(i).Split("|").GetValue(0)**

**Dim StrValue As String = IngFields(i).Split("|").GetValue(1)**

**Select Case StrDataType.ToUpper**

**Case Is = "INT"**

**ComIngres.Parameters.Add("Parameter" & i, IngresType.Int).Value = CInt(StrValue)**

**Case Is = "VAR"**

**ComIngres.Parameters.Add("Parameter" & i, IngresType.VarChar).Value = StrValue**

**Case Is = "CHR"**

**ComIngres.Parameters.Add("Parameter" & i, IngresType.Char).Value = StrValue**

**Case Is = "DTE"**

**ComIngres.Parameters.Add("Parameter" & i, IngresType.Date).Value = CDate(StrValue)**

**Case Is = "MON"**

**ComIngres.Parameters.Add("Parameter" & i, IngresType.Double).Value = CDbl(StrValue)**

**End Select**

**Next**

**dr.SelectCommand = ComIngres**

**dr.Fill(OutputTable)**

**Catch x As Exception**

**Console.Write(x.Message.ToString)**

**ComIngres.Dispose()**

**Finally**

**ConnIngres.Close()**

**ConnIngres.Dispose()**

**dr.Dispose()**

**ComIngres.Dispose()**

**End Try**

**For x = 0 To OutputTable.Columns.Count - 1**

**OutputTable.Columns(x).ColumnName = "result element" & x**

**Next**

**Return OutputTable**

**End Function**

**#End Region**

**#Region "Generate Number Max"**

**Public Function IngDataTableGenNum(ByVal ProcedureName As String, ByVal ParamArray IngFields() As String) As DataTable**

**Dim ConnIngres As New IngresConnection**

**Dim ComIngres As New IngresCommand**

**Dim dr As New IngresDataAdapter**

**Dim OutputTable As New DataTable**

**Try**

**ConnIngres.ConnectionString = IngConn1**

**ConnIngres.Open()**

**ExecuteReadLockSession(ConnIngres)**

**With ComIngres**

**.Connection = ConnIngres**

**.Parameters.Clear()**

**.CommandText = "sp\_menu\_sel1"**

**.CommandType = Data.CommandType.StoredProcedure**

**End With**

**For i = 0 To UBound(IngFields)**

**Dim StrDataType As String = IngFields(i).Split("|").GetValue(0)**

**Dim StrValue As String = IngFields(i).Split("|").GetValue(1)**

**Select Case StrDataType.ToUpper**

**Case Is = "INT"**

**ComIngres.Parameters.Add("Parameter" & i, IngresType.Int).Value = CInt(StrValue)**

**Case Is = "VAR"**

**ComIngres.Parameters.Add("Parameter" & i, IngresType.VarChar).Value = StrValue**

**Case Is = "CHR"**

**ComIngres.Parameters.Add("Parameter" & i, IngresType.Char).Value = StrValue**

**Case Is = "DTE"**

**ComIngres.Parameters.Add("Parameter" & i, IngresType.Date).Value = CDate(StrValue)**

**Case Is = "MON"**

**ComIngres.Parameters.Add("Parameter" & i, IngresType.Double).Value = CDbl(StrValue)**

**End Select**

**Next**

**dr.SelectCommand = ComIngres**

**dr.Fill(OutputTable)**

**Catch x As Exception**

**Console.Write(x.Message.ToString)**

**ComIngres.Dispose()**

**Finally**

**ConnIngres.Close()**

**ConnIngres.Dispose()**

**dr.Dispose()**

**ComIngres.Dispose()**

**End Try**

**For x = 0 To OutputTable.Columns.Count - 1**

**OutputTable.Columns(x).ColumnName = "result element" & x**

**Next**

**Return OutputTable**

**End Function**

**#End Region**

**End Module**

A screenshot of a computer

Description automatically generated